35 U.S.C. §102

Goettelmann

Claims 1 – 15 are rejected under 35 U.S.C. §102(b) as being anticipated by Goettelmann et al., U.S. Patent No. 5,313,614 (hereinafter, Goettelmann). The rejection is respectfully traversed.

Claim 15

Claim 15 recites:

> In a computer which translates instructions from a target instruction set to a host instruction set, a method for determining validity of a translated instruction comprising:
> testing a memory address of a target instruction to be executed against a memory address associated with a translation of a target instruction, wherein said translation is linked to a translation of another target instruction;
> executing said translation if said memory addresses compare; and
> generating an exception if said memory addresses do not compare.

Claim 15 recites that a memory address of a target instruction to be executed is tested against a copy of the memory address of the target instruction from which a translation of the target instruction was made. Moreover, Claim 15 recites that the translation is linked to a translation of another target instruction. It is respectfully submitted that Goettelmann fails to disclose these claimed limitations.

Embodiments according to the present invention link the translation of one instruction to the translation of another instruction. For example, Figure 4 of the Specification illustrates different embodiments for linking a first translation to a second translation. In each embodiment, there is a translation of a target instruction

(e.g., one of the second translations) linked to an earlier translation (e.g., the first translation).

The rejection cites Goettelmann at col. 9, lines 43 et seq. as teaching a mapping action for determining the validity of memory addresses of the translated instructions. In this passage Goettelmann is describing an emulation process comprising decoding instructions, mapping the decoded instructions to equivalent instructions of the target machine, and executing the instructions (col. 9, lines 29-34). Applicants note that Goettelmann uses the term "target machine" to mean the machine to which instructions are translated, whereas the preamble to Claim 15 reads, "a computer which translates instructions from a target instruction set to a host instruction set."

Goettelmann goes on to define "mapping" as the action of determining, for each source machine instruction, an equivalent sequence of operations in the target machine, using the target machine instruction set (col. 9, lines 44-46). Applicants do not understand this determination to be "testing a memory address of a target instruction to be executed against a memory address associated with a translation of a target instruction, wherein said translation is linked to a translation of another target instruction", as claimed. For example, Applicants do not understand Goettelmann to disclose testing a memory address in this portion of the passage.

The passage on the definition of mapping continues by stating that the emulator adds an aforementioned offset to the addresses contained in the various

instructions, so that the instructions in the software emulator refer to the appropriate locations in the target machine RAM (col. 9, lines 46-51). The offset being referred to is described at col. 8, line 5-29. For example, if the address 0 appears in a host instruction, it is converted into 0xc000 in the corresponding emulated/translated instruction (col. 9, lines 19-22). See also Figure 4 illustrating the target and source machine address spaces. Applicants do not understanding the adding of this offset to be "testing a memory address of a target instruction to be executed against a memory address associated with a translation of a target instruction, wherein said translation is linked to a translation of another target instruction," as claimed. For example, Applicants do not understand Goettelmann to disclose testing a memory address in this portion of the passage.

For the foregoing rationale, Goettelmann fails to disclose the limitations of Claim 15. As such, allowance of Claim 15 is respectfully solicited.

Claims 1 and 8

Claim 1 recites:

> In a computer which translates instructions from a target instruction set to a host instruction set, a method for determining validity of a translation of a target instruction linked to an earlier translation comprising the steps
> testing a memory address of a target instruction to be executed against a copy of the memory address of the target instruction from which a translation of the target instruction was made,
> executing the translation if the addresses compare, and
> generating an exception if addresses do not compare.

Claim 1 recites a method for determining validity of a translation of a target instruction linked to an earlier translation. For the reasons discussed in the response to Claim 15, it is respectfully submitted that Goettelmann fails to disclose

determining validity of a translation of a target instruction linked to an earlier translation.

For the foregoing rationale, it is respectfully submitted that Claim 1 is not anticipated by Goettelmann. Claim 8 recites similar limitations. As such, Claim 8 is not anticipated by Goettelmann.

Claims 2 and 9

Claim 2 recites:

> A method as claimed in Claim 1 in which the step of testing a memory address of a target instruction to be executed against a copy of the memory address of the target instruction from which a translation of the target instruction was made is a process separate from the translation of the target instruction.

In Claim 2, the testing of the memory address is a process separate from the translation of the target instruction. For example, when it is not known when the translation is made whether it will be linked to another translation or not, then the prologue process is generated as a separate short prologue when the linking of the two translations occurs. At this time, the preceding translation is provided a jump instruction to the prologue process; and the prologue completes (if the test is met) with another jump instruction to the succeeding translation. Figure 4 of the Specification illustrates embodiments having a first translation that is linked to a second translation. In one embodiment, the lower of the two second translations does not contain the prologue as a part of the translation. In this embodiment, the jump from the first translation goes to the prologue. The prologue has a jump that goes to the second translation, in this embodiment.

Goettelmann fails to disclose testing a memory address of a target instruction to be executed as a part of a process that is separate from the translation of the target instruction. The rejection cites col. 9, lines 58-62. Applicants do not understand this passage to disclose the claimed limitation. The cited passage describes how Goettelmann proceeds if upon decoding an instruction, the emulator determines that it is a system call. In this case, the "system simulation 57 thereupon carries out the requested service by accessing target machine address space 55 either directly of via target machine system software 58." Goettelmann does not expressly disclose in this passage "a testing of a memory address of a target instruction," as claimed. Moreover, Applicants do not understand Goettelmann to disclose the limitations of Claim 2 elsewhere.

For the foregoing rationale, it is respectfully submitted that Claim 2 is not anticipated by Goettelmann. Claim 9 recites similar limitations. As such, Claim 9 is not anticipated by Goettelmann.

Claims 3 and 10

Claim 3 recites:

> A method as claimed in Claim 1 in which the step of testing a memory address of a target instruction to be executed against a copy of the memory address of the target instruction from which a translation of the target instruction was made is included as a part of the translation of the target instruction.

Claim 3 recites that the testing of the memory address is included as a part of the translation of the target instruction. For example, if it is known that the

translation will be linked to a previous translation when it is translated and a check of the physical address is required, then the prologue process (described and illustrated in Figure 3) may be included in the translation. In such as case, the preceding translation merely jumps to the next translation where the address consistency process of the prologue is executed before the translation is executed. Figure 4 of the Specification illustrates embodiments having a first translation that is linked to a second translation. In one embodiment, the upper of the two second translations contains a prologue as a part of the translation. In this embodiment, the jump from the first translation goes to the second translation.

Goettelmann fails to disclose testing the memory address as a part of the translation of the target instruction. The rejection cites col. 20, lines 62-66. Applicants do not understand this passage to disclose the claimed limitation. For example, Applicants do not understand Goettelmann to disclose a testing of a memory address, as claimed. In contrast, the cited passage is part of a description of step 1335 of Figure 13 and relating to a mapping process. Goettelmann from col. 19, line 1 – col. 21, line 15 describes this mapping process with the emphasis on dealing with side effects. An example of a side effect is the setting of a condition code bit (col. 19, lines 34-36). The mapping process includes factoring in known side during a translation. For example, a skeletal intermediate language code sequence is developed for each source machine instruction factoring in primary effects and known side effects (col. 20, lines 21-26). The mapping is disclosed as consisting of adding to the already created intermediate language code 1) code for fetching operands, 2) the

appropriate skeleton, 3) code for storing the result of the instruction, if necessary (col. 20, lines 62-66). However, Goettelmann fails to disclose, "testing the memory address as a part of the translation of the target instruction," as claimed. Moreover, Applicants do not understand Goettelmann to disclose the limitations of Claim 3 elsewhere.

For the foregoing rationale, it is respectfully submitted that Claim 3 is not anticipated by Goettelmann. Claim 10 recites similar limitations. As such, Claim 10 is not anticipated by Goettelmann

Claims 4 and 11

Claim 4 recites:

> A method as claimed in Claim 1 which includes an additional step of copying a memory address of a target instruction when a translation of the target instruction is made and linked to an earlier translation.

Claim 4 recites that the memory address of a target address is copied when a translation of the target instruction is made and linked to an earlier translation.

Goettelmann fails to disclose "copying a memory address of a target instruction when a translation of the target instruction is made and linked to an earlier translation," as claimed. The rejection cites col. 30, lines 6-12. Applicants do not understand this passage to disclose the claimed limitation. Rather, the passage relates to binding a translated application, the translated system software, and target machine system software (col. 29, lines 66-68). The binding comprises linking and loading. The linking comprises converting symbolic references in the aforementioned code to numerical offsets that are relative to the starting address of

the combined code. This allows the translated code to be executed within the target machine. However, Applicants do not understand the code from which the translation was made to be included in the bind. For example, Figures 5 and 6 of Goettelmann illustrate the source machine application 51 and 61, respectively. Figure 6 clearly depicts the source machine application 61 being translated to produce a translated application 63. Applicants do not understand the source machine application to be in the bind.

Applicants note that Claim 4 recites that the address that is copied is that of the target instruction. The preamble of Claim 1, from which Claim 4 depends, recites "in a computer which translates instructions from a target instruction set to a host instruction set." Thus, the target instruction set is the set of instructions to be translated. The cited passage of Goettelmann refers to binding the translated application, the translated system software and target machine system software. As Goettelmann does not disclose putting the source machine application in the binding process, Applicants do not understand Goettelmann to disclose the "copying a memory address of a target instruction when a translation of the target instruction is made and linked to an earlier translation," as claimed.

For the foregoing rationale, it is respectfully submitted that Claim 4 is not anticipated by Goettelmann. As such, allowance of Claim 4 is respectfully submitted. Claim 11 recites similar limitations. As such, allowance of Claim 11 is respectfully submitted.

Claims 5-7 and 12-14 depend from Claim 1 and 8, which are respectfully believed to be allowable over Goettelmann. As such, Claims 5-7 and 12-14 are respectfully believed to be allowable over Goettelmann.

De Nicolas

Claims 1-2, 5-9, and 12-14 are rejected under 35 U.S.C. §102(b) as being anticipated by de Nicolas et al., U.S. Patent No. 5,167,023 (hereinafter, de Nicolas). The rejection is respectfully traversed.

Claims 1 and 8

Claim 1 recites a method for determining validity of a translation of a target instruction linked to an earlier translation. It is respectfully submitted that de Nicolas fails to disclose determining validity of a translation of a target instruction linked to an earlier translation. Moreover, Claim 1 recites that the memory address of the target instruction is tested against a copy of the memory address of the target instruction from which a translation of the target instruction was made. It is respectfully submitted that de Nicolas fails to disclose this limitation of Claim 1.

For the foregoing rationale, it is respectfully submitted that Claim 1 is not anticipated by de Nicolas. As such, allowance of Claim 1 is respectfully submitted. Claim 8 recites similar limitations. As such, allowance of Claim 8 is respectfully submitted.

Claims 2 and 9

In Claim 2, the testing of the memory address is a process separate from the translation of the target instruction. For example, when it is not known when the translation is made whether it will be linked to another translation or not, then the prologue process is generated as a separate short prologue when the linking of the two translations occurs. At this time, the preceding translation is provided a jump instruction to the prologue process; and the prologue completes (if the test is met) with another jump instruction to the succeeding translation. Figure 4 of the Specification illustrates embodiments having a first translation that is linked to a second translation. In one embodiment, the lower of the two second translations does not contain the prologue as a part of the translation. In this embodiment, the jump from the first translation goes to the prologue. The prologue has a jump that goes to the second translation, in this embodiment.

De Nicolas fails to disclose testing a memory address of a target instruction to be executed as a part of a process that is separate from the translation of the target instruction. For support, the rejection cites de Nicolas at col. 5, lines 59-63, and col. 6, lines 1-7. The cited passages disclose a way to reduce the number of host instructions per simulated instructions for the case when an instruction stores to memory. In particular, the passages disclose that when an instruction updates memory, a determination is made as to whether the update is to a video buffer or to a subsequent instruction. However, Applicants do not understand the cited passage to disclose the testing of a memory address of a target instruction to be executed as a part of a process that

is separate from the translation of the target instruction. Moreover, Applicants do not understand de Nicolas to disclose the limitations of Claim 2 elsewhere.

For the foregoing rationale, it is respectfully submitted that Claim 2 is not anticipated by de Nicolas. Claim 9 recites similar limitations. As such, Claim 9 is not anticipated by de Nicolas.

Claims 5-7 and 12-14 depend from Claim 1 and 8, which are respectfully believed to be allowable over de Nicolas. As such, of Claims 5-7 and 12-14 are respectfully believed to be allowable over de Nicolas.

Fogg

Claims 1-3, 5-10, and 12-14 are rejected under 35 U.S.C. §102(b) as being anticipated by Fogg Jr. et al., U.S. Patent No. 4,951,195 (hereinafter, Fogg). The rejection is respectfully traversed.

Claims 1 and 8

Claim 1 recites a method for determining validity of a translation of a target instruction linked to an earlier translation. It is respectfully submitted that Fogg fails to disclose determining validity of a translation of a target instruction linked to an earlier translation. Moreover, Claim 1 recites that the memory address of the target instruction is tested against a copy of the memory address of the target instruction from which a translation of the target instruction was made. It is respectfully submitted that Fogg fails to disclose this limitation of Claim 1.

For the foregoing rationale, it is respectfully submitted that Claim 1 is not anticipated by Fogg. Claim 8 recites similar limitations. As such, Claim 8 is not anticipated by Fogg.

Claims 2 and 9

In Claim 2, the testing of the memory address is a process separate from the translation of the target instruction. For example, when it is not known when the translation is made whether it will be linked to another translation or not, then the prologue process is generated as a separate short prologue when the linking of the two translations occurs. At this time, the preceding translation is provided a jump instruction to the prologue process; and the prologue completes (if the test is met) with another jump instruction to the succeeding translation. Figure 4 of the Specification illustrates embodiments having a first translation that is linked to a second translation. In one embodiment, the lower of the two second translations does not contain the prologue as a part of the translation. In this embodiment, the jump from the first translation goes to the prologue. The prologue has a jump that goes to the second translation, in this embodiment.

Fogg fails to disclose testing a memory address of a target instruction to be executed as a part of a process that is separate from the translation of the target instruction. The rejection cites col. 11, lines 45-55. Applicants do not understand this passage to disclose the claimed limitation. The first sentence of the cited passage states that translations are saved such that they may be re-used if the processor transfers control to the same address again. However,

nothing is disclosed here regarding the testing of a memory address of a target instruction to be executed as a part of a process that is separate from the translation of the target instruction, as claimed. The cited passage goes on to disclose how the next instruction pointer (IP) is determined for instructions that do not transfer control. However, again there is no disclosure as to any testing of a memory address of a target instruction to be executed, as claimed. Moreover, Applicants do not understand Fogg to disclose the limitations of Claim 2 elsewhere.

For the foregoing rationale, it is respectfully submitted that Claim 2 is not anticipated by Fogg. Claim 9 recites similar limitations. As such, Claim 9 is not anticipated by Fogg.

Claims 3 and 10

Claim 3 recites that the testing of the memory address is included as a part of the translation of the target instruction. For example, if it is known that the translation will be linked to a previous translation when it is translated and a check of the physical address is required, then the prologue process (described and illustrated in Figure 3) may be included in the translation. In such as case, the preceding translation merely jumps to the next translation where the address consistency process of the prologue is executed before the translation is executed. Figure 4 of the Specification illustrates embodiments having a first translation that is linked to a second translation. In one embodiment, the upper of the two second translations contains a prologue as a part of the translation. In this embodiment, the jump from the first translation goes to the second translation.

Fogg fails to disclose testing the memory address as a part of the translation of the target instruction, as claimed. The rejection cites col. 15, line 56 – col. 16, line 34, and Figure 9. Figure 9 depicts a way to map memory using a first shared memory segment 190 and a second shared memory segment 180. The memory mapping is used to translate addresses from the first processing system into the addresses of the second processing system (col. 15, lines 16-20). The first shared memory segment 190 stores an image of the memory of the first processing system and a second shared memory segment 180 indicates for each memory location, the contents contained in that image.

Figure 9 of Fogg shows translated instructions 157 in the first shared memory segment 190. However, Fogg does not show "testing of the memory address is included as a part of the translation of the target instruction," as claimed. Moreover, Applicants do not understand Fogg to disclose the limitations of Claim 3 elsewhere.

For the foregoing rationale, it is respectfully submitted that Claim 3 is not anticipated by Fogg. Claim 10 recites similar limitations. As such, Claim 10 is not anticipated by Fogg.

Claims 5-7 and 12-14 depend from Claim 1 and 8, which are respectfully believed to be allowable over Fogg. As such, Claims 5-7 and 12-14 are respectfully believed to be allowable over Fogg.

For the foregoing rationale, Claims 1, 2, 5-7, and 12-14 are not anticipated by Goettelmann, de Nicolas, or Fogg. As such, allowance of Claims 1, 2, 5-7, and 12-14 is respectfully solicited. For the foregoing rationale, Claim 3 is not anticipated by Goettelmann or Fogg. As such, allowance of Claim 3 is respectfully solicited. For the foregoing rationale, Claims 4 and 15 are not anticipated by Goettelmann. As such, allowance of Claims 4 and 15 is respectfully solicited.

CONCLUSION

In light of the above listed remarks, reconsideration of the rejected Claims is requested. Based on the arguments presented above, it is respectfully submitted that Claims 1-15 overcome the rejections of record and, therefore, allowance of Claims 1-15 is earnestly solicited.

Should the Examiner have a question regarding the instant response, the Applicants invite the Examiner to contact the Applicants' undersigned representative at the below listed telephone number.

Respectfully submitted,
WAGNER, MURABITO & HAO LLP

Dated: 2/18, 2003

Ronald M. Pomerenke
Registration No. 43,009

Address: WAGNER, MURABITO & HAO LLP
Two North Market Street
Third Floor
San Jose, California 95113

Telephone: (408) 938-9060 Voice
(408) 938-9069 FAX